浅谈媒体协同数据库设计

摘　要：数据库是信息系统的核心和基础，一个好的数据库设计在实现信息系统方便、及时、准确地获得所需的信息之外，还要满足易维护、易扩充等要求，并考虑到数据的一致性、冗余性、访问效率等。

关键词：数据库；表；检索；索引
中图分类号：G202　　文献标识码：A
文章编号：1671-0134（2017）08-081-02　　DOI：10.19483/j.cnki.11-4653/n.2017.08.029

■文 / 宋凌宇　张树勇　阚景森

引言

2015 年 3 月起，随着国家数字复合出版系统工程——媒体协同（即 07 包）系统的设计和研发逐步展开，数据库的设计变得尤为关键。通过对以往一些应用系统开发过程出现的问题进行分析，然后规避这些问题，设计出一个规范、高性能、易维护、易扩展的数据库，以实现 07 包的功能需求及后期的扩展。

1. 以往应用系统的问题

笔者参与过一些应用系统的设计和研发工作，经历过项目研发过程中的瓶颈和问题，这些问题表现在以下几个方面。

1.1 数据表和字段命名混乱

有些项目的数据表和字段命名不规范，往往难以找到所需要的库表，给开发人员带来很多不便，甚至困惑。

1.2 难扩展功能

有些项目的数据表字段数量刚好符合开始设计时的功能需求，那么在开发过程中有功能扩充时，就要增加数据表字段、数据表，有可能会造成程序的修改，甚至重新开发，这样就会加大开发人员工作量，也有可能造成项目的延期；再者项目交付多个用户后，会造成项目多版本维护，增加维护的难度。

2. 数据检索慢

有些项目在设计之初，考虑得相当完善，规避了许多数据库设计时的问题。但随着用户使用数据的增涨，尤其达到百万级以上，数据的检索速度明显变慢，势必给用户带来不便，甚至厌烦。

如：部分用户应用系统情况

用户	稿件表记录	日志表记录	检索耗时
半岛都市报	30 多万条	180 多万条	5 分钟
大众日报	40 多万条	200 多万条	5 分钟
…			

3.（07 包）数据库设计

数据库的设计是 07 包的各功能能否紧密地结合在一起以及如何结合的关键所在，是 07 包开发和建设的重要组成部分。从以下几个方面对数据库的设计进行规范和优化设计。

3.1 命名规范

表：T+ 代表字母 + 表名 。数据表名由具有能描述表内容等特殊含义的单词或缩写组成。如：TISTORYSOURCE 稿件来源表。

字段：类型前缀 + 字段名。必须以有特征含义的单词或缩写组成。

主键：PK_。主键名称应是 前缀 + 表名 + 构成的字段名。如果复合主键的构成字段较多，则只包含第一个字段。表名可以去掉前缀。

外键：FK_。外键名称应是 前缀 + 外键表名 + 主键表名 + 外键表构成的字段名。表名可以去掉前缀。

索引：IDX_。索引名称应是 前缀 + 表名 + 构成的字段名。如果复合索引的构成字段较多，则只包含第一个字段，并添加序号。表名可以去掉前缀。

3.2 逻辑分块

根据 07 包功能数据的逻辑进行分块设计。

TI_（TableInfo_ 缩写）。 是跟系统中参数设置相关的数据表，如：稿件体裁、来源、分类等。

TB_（TableBase_ 缩写）。 是跟系统中基础信息设置相关的数据表， 如：部门、人员、作者、角色、共享范围等。

TW_（TableWork _ 缩写）是跟 07 包实际业务相关的数据表，如：稿件、报题、任务、选题等。

TR_(TableRole_ 缩写)。是跟角色权限相关的数据表。

3.3 数据关联设计

应 07 包业务要求，有很多关联查询需求，在设计之初就有必要正确处理多对多的关系。尽量消除多对多情况，将一个多对多的关系，变为两个一对多的关系。

基金项目：本文受国家数字复合出版系统工程 - 媒体协同工作系统 (1141STC40519/07) 项目资助。

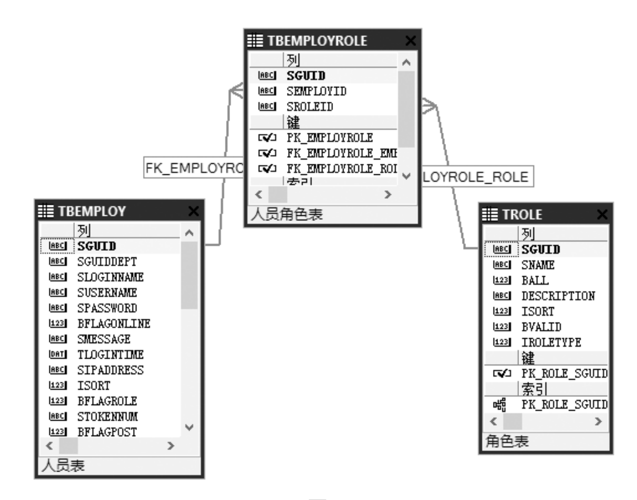


图 1

如图 1：一个人员有可能有多个角色，反之一个角色赋值多个人员。在角色和人员表中增加一个角色人员表，就变为两个一对多的关系了。

再有，因用户个性需求而易改变的关键数据，增加“名值表”设计。“名值表”，顾名思义就是，那些键被其他数据关联着的值修改时，不会造成数据存放的混乱，便于多表关联查询。

第三，为实现数据的完整性，在设计数据表时就考虑了数据冗余，以及事务的添加、级联删除和级联更新等。

3.4 超前设计

07 包数据库的设计不但满足当前的功能需求，还要考虑未来功能的扩展，增加 07 包的灵活性。

预留数据表。在设计时考虑未来可能的业务，预设计一些数据表。如：音视频信息扩展表，拟记录音视频的码率、时长等信息。

预留字段。在设计时针对系统业务数据表，基本都预留 1–2 个字段，便于功能的扩展和后续的业务拓展。如果这些预留字段不能满足未来业务的需求，那么就通过添加数据表，实现更高的可扩充性要求。

3.5 优化设计

对于大的数据库表，合理的索引能够提高整个数据库的操作效率。索引在数据库优化中占有一个非常大的比例，在设计数据表时，充分考虑到当单表数据量很大时，比如说是百万数量级，如果我们使用普通的查询语句，耗时会非常多。给这些表建上好的索引，能将检索效率提高几十甚至几百倍。

在设计索引时，遵循以下规则：

①主键（sguid）的数据列、有外键的数据列（fk_sguid）一定要建立索引。

②对于经常查询的数据列根据需要建立索引。

③对于需要在指定范围内快速或频繁查询的数据列，如“**name LIKE ‘a%’”，最好建立索引。

④经常用在 WHERE 子句中的数据列，建立索引。

⑤经常出现在关键字 order by、group by、distinct 后面的字段，建立索引。如果建立的是复合索引，索引的字段顺序要和这些关键字后面的字段顺序一致，否则索引不会被使用。

⑥对于那些查询中很少涉及的列，重复值比较多的列不要建立索引。

⑦对于定义为 text、image、blob 和 bit 数据类型的列不要建立索引。

⑧对于经常存取的列避免建立索引 。

⑨限制表上的索引数目。对一个存在大量更新操作的表，所建索引的数目一般不要超过 3 个，最多不要超过 5 个。索引虽说提高了访问速度，但太多索引会影响数据的更新操作。

⑩对复合索引，按照字段在查询条件中出现的频度建立索引。在复合索引中，记录首先按照第一个字段排序。对于在第一个字段上取值相同的记录，系统再按照第二个字段的取值排序，以此类推。只有复合索引的第一个字段出现在查询条件中，该索引才可能被使用，因此将应用频度高的字段，放置在复合索引的前面，会使系统最大可能地使用此索引，发挥索引的作用。

⑪在 Join 查询时，两个表中 Join 的字段建立索引。

遵循以上规则对 07 包中稿件表和日志表、操作历史表等关键列增加索引设置。

附：用户数据库优化后应用系统检索情况，如下表：

用户	稿件表记录	日志表记录	索引前检索耗时	索引后检索耗时	备注
半岛都市报	30 多万条	180 多万条	5 分钟	1–3 秒	
大众日本	40 多万条	200 多万条	5 分钟	1–3 秒	服务器性能好
…					

4. 总结

本文通过论述一些应用系统的现存问题，得出数据库设计的重要性。提出 07 包的数据库设计时要遵循的原则和规范，从而设计出一个满足 07 包功能需求和未来扩展的高性能、易维护、易扩展的数据库。媒

参考文献

[1] 励文杰 . 大型数据库 ORACLE 数据库的优化设计方案 [J]. 科技风，2011（19）.

[2] 赵静宇 . 数据库设计规范化的理论研究与应用 [J]. 电子技术与软件工程，2013（21）.

[3] 汪辉 . 基于大数据应用系统架构的设计与实现 [J]. 电子技术与软件工程，2015（20）.

（作者单位：潍坊北大青鸟华光照排有限公司；东营日报社）